الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
وزارة التكوين والتعليم المهنيين
Ministère de la Formation et de l'Enseignement Professionnels

Centre National de l'Enseignement Professionnel à Distance (CNEPD)

المركز الوطني للتعليم المهني عن بعد
(م.و.ت.م.ب)

COURS DE MATHÉMATIQUES

SÉRIE 03

OBJECTIF PÉDAGOGIQUE : À la fin de cette série, le stagiaire doit être capable d'appliquer les portes logiques et circuits logiques.

PLAN DE LA LEÇON :

INTRODUCTION

I- DUALITÉ, THÉORÈMES FONDAMENTAUX

II- ORDRE ET ALGÈBRE DE BOOLE

III- EXPRESSION BOOLÉENNE, FORME POLYGONALE

IV- PORTES LOGIQUES, CIRCUITS LOGIQUES

V- SIMPLIFICATION DES CIRCUITS LOGIQUES :
DIAGRAMME DE KARNAUGH DIAGRAMME DE VENN

INTRODUCTION :

Définit en 1847 par Georges Boole (1815-1864), physicien Anglais

- Algèbre applicable au raisonnement logique qui traite des fonctions à variables binaires (deux valeurs).
- Ne s'applique pas aux systèmes à plus de deux états d'équilibre.
- Permet d'étudier les circuits logiques (un système logique sert à modifier des signaux).

L'algèbre de Boole permet de manipuler des valeurs logiques

- Une valeur logique n'a que deux états possibles :
- Vraie(1) ou Fausse(0).
- Plusieurs valeurs logiques peuvent être combinées pour donner un résultat qui est lui aussi une valeur logique

Exemple :

- Arrêt marche
- Ouvert fermé
- Enclenché déclenché
- Avant arrière
- Vrai faux
- Conduction blocage

I - DUALITÉ, THÉORÈMES FONDAMENTAUX :

1- Définition d'Algèbre de Boole :

Soit B, un ensemble ayant au moins 2 ´éléments notes 0 et 1. On munit B de 2 opérations binaires internes, notées« + » et « • », et d'une opération unaire notée x → bar (x) = $\overline{x}$ (B, +, •, bar) possède une structure d'Algèbre de BOOLE, si les opérations ont les propriétés suivantes :

- **Commutativité : a + b = b + a et a • b = b • a**
- **Associativité** : **« + »** et **« • »**sont associatives, c'est `a dire **:**

$$(\forall a \in B)\,(\forall b \in B)\,(\forall c \in B)\,(a+b)+c = a+(b+c) = a+b+c$$

$$(\forall a \in B)\,(\forall b \in B)\,(\forall c \in B)\,(a \bullet b) \bullet c = a \bullet (b \bullet c) = a \bullet b \bullet c$$

Éléments neutres : 0 est le neutre pour + et 1 est le neutre pour •, c'est `a dire :

$$(\forall a \in B)\, a+0 = 0+a = a \quad \text{et} \quad (\forall a \in B)\, a \bullet 1 = 1 \bullet a = a$$

- **Distributivité :** Chaque opération est distributive par rapport à l'autre **:**

$$(\forall a \in B)\,(\forall b \in B)\,(\forall c \in B)\, a \bullet (b+c) = a \bullet b + a \bullet c$$

$$(\forall a \in B)\,(\forall b \in B)\,(\forall c \in B)\, a + (b \bullet c) = (a+b) \bullet (a+c)$$

2-Le théorème de dualité :

En algèbre de BOOLE, tout théorème se présente sous 2 formes duales. L'énonce du théorème dual s'obtient en permutant systématiquement dans l'énonce du théorème initial, les « + »par les « • » et les neutres 0 par les 1.

2.1- Pratique du calcul BOOLEEN :

- **Proposition** :

1. Nous avons : 1 = 0 et 0 = 1
2. Nous avons de plus : a = a
3. Principe d'idempotence : a + a = a et a • a = a
4. «1» est absorbant pour l'addition, et 0 pour la multiplication i.e.

1 + a = 1 et 0 • a = 0

- **Démonstration** :

1. D'après l'axiome 5, $0+\overline{0}=1$, comme 0 est le neutre pour +, c'est fini ; Et ce n'est pas plus difficile pour l'autre !! *(principe de dualité)*
2. $\overline{a}\overline{\overline{a}}=0$, et $a\overline{a}=0$, et $\overline{a}+\overline{\overline{a}}=1$, et $a+\overline{a}=1$; de l'unicité du complémentaire on en déduit que $\overline{\overline{a}}=a$
3. De $a\overline{a}=0$, on tire $a\overline{a}+a=a$ et, $a\overline{a}+a=a+a\overline{a}=(a+a)\bullet(a+\overline{a})=(a+a)\bullet 1=a+a$; de la distributivité de + sur•, on trouve :

$$a\overline{a}+a=a+a\overline{a}=(a+a)\bullet(a+\overline{a})=(a+a)\bullet 1=a+a$$

   D'où $a+a=a$
4. L'égalité $a\bullet a=a$ est la duale de la précédente.

2.2 -<u>Démonstration de 1 est absorbante pour l'addition</u> :

- **Théorèmes fondamentaux :**

$$1+a=(a+\overline{a})+a=(a+a)+\overline{a}=a+\overline{a}=1$$

- **Théorème d'absorption :**

Dans une somme de produits booléens, un terme absorbe ses multiples, c'est à dire :

$$a + ab = a$$

- **Démonstration :**

C'est simple! $a + ab = a \bullet 1 + ab = a (1 + b) = a (1) = a$

2.3- <u>Théorème de redondance</u> :

$$(\forall a\in B)\,(\forall b\in B)\,(\forall c\in B)\,,\quad \overline{a}x+ay=\overline{a}x+ay+xy$$

- **Démonstration :**

$$\overline{a}x+ay=[\overline{a}x+(\overline{a}x)\,y]+[ay+(ay)\,x]$$

Car tout nombre absorbe ses multiples

$$= \overline{a}\,(xy) + a\,(xy) + \overline{a}x + ay$$

Associativité et commutativité des opérations, et réorganisation des calculs

$$= \overline{a}x + ay + xy\,(a + \overline{a}) == \overline{a}x + ay + xy$$

3-<u>Théorème de MORGAN</u> :

$$(\forall a \in B)\,(\forall b \in B)\; \overline{a+b} = \overline{a} \bullet \overline{b} \; et \; \overline{a \bullet b} = \overline{a} + \overline{b}$$

Les propositions sont ”duales” l’une de l’autre

Démonstration :

$$\overline{a} \bullet \overline{b} + (a + b) = \overline{a} \bullet \overline{b} + a \bullet 1 + b$$

En utilisant le principe de redondance2.3, o`u nous pouvons écrire :
$\overline{a}\overline{b} + a \bullet 1 = \overline{a}\overline{b} + a \bullet 1 + \overline{b} \bullet 1$

dire que nous avons choisi dans 2.3 $x = \overline{b}$ et $y = 1$; et nous avons donc :

$$\overline{a} \bullet \overline{b} + a \bullet 1 + b = \overline{a} \bullet \overline{b} + a \bullet 1 + \overline{b} \bullet 1 + b = \overline{a} \bullet \overline{b} + a + \overline{b} + b = \overline{a} \bullet \overline{b} + a + 1 = 1$$

Donc, $\overline{a+b} = \overline{a} \bullet \overline{b}$

II- ORDRE ET ALGÈBRE DE BOOLE :

1- Algèbre de Boole comme structure ordonnée :

Une algèbre de Boole est un ensemble ordonné $(E, \leq)$:

- Il existe un plus petit élément, noté dans la suite 0 (il est souvent noté également $\perp$), et un plus grand élément noté dans la suite 1 (il est souvent noté également $\top$) ;
- Deux éléments a, b de E ont une borne supérieure, notée dans la suite $a \vee b$, et une borne inférieure notée dans la suite $a \wedge b$ (c'est-à-dire que c'est un treillis) ;
- L'opération $\wedge$ est distributive sur l'opération $\vee$, et l'opération $\vee$ est distributive sur l'opération $\wedge$, c'est-à-dire que pour tous a, b et c de E :

  $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$,

  $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$;

- Pour tout élément a de E, il existe un élément a' de E, appelé *complément* de E et vérifiant :

  $a \wedge a' = 0$ et $a \vee a' = 1$.

Ainsi une algèbre de Boole est un treillis distributif, borné (avec plus petit et plus grand élément), et complémenté. Il suffirait de donner une des lois de distributivité, l'une entraînant l'autre dans un treillis (voir l'article lié).

On montre que chaque élément possède un unique complément. En effet si a^1 et a' sont des compléments de a, en utilisant les propriétés des compléments, de l'ordre, et la distributivité on obtient :

$$a^1 = (a \wedge a') \vee a^1 = (a \vee a^1) \wedge (a' \vee a^1) = a' \vee a^1$$

C'est-à-dire qu'a1 $\leq a'$. De la même façon $a^1 = a' \wedge a^1$, donc $a' \leq a^1$, d'où l'égalité.

Par unicité du complément, l'application qui à *a* associe sont complément est involutive (*a’’* = *a*). Elle échange *0* et *1*.
Également par unicité du complément et par distributivité on montre que le passage au complément échange ∧ et ∨, ce sont les lois de De Morgan (voir paragraphe suivant).

On déduit des axiomes d'ordres que les opérations ∧ et ∨ sont associatives, commutatives, que 0 est neutre pour ∨ et absorbant pour ∧, que 1 est neutre pour ∧ et absorbant pour ∨, et que ces opérations sont idempotentes.

III- EXPRESSION BOOLÉENNE, FORME POLYGONALE :

Une expression booléenne est une expression qui a pour valeur le type de données booléen. Les expressions **Booléen** peuvent revêtir plusieurs formes. La forme la plus simple est la comparaison directe de la valeur d'une variable **Booléen** à un littéral **Booléen**, comme le montre l'exemple suivant :

Fonction booléenne de trois variables a, b, c :

Écrivez une fonction des trois variables booléennes a, b et c.
Les variables sont 'a', 'b' et 'c'.
Les symboles de constantes sont '1' ou 'V' pour Vrai, '0' ou 'F' pour Faux.
'!' est l'opérateur unaire de complémentation (!a est le complément de a, on peut aussi utiliser les signes ?, -, /, \, N, n).
'+' '.' Sont les opérateurs binaires de la somme et du produit booléens, le signe '.' peut être omis ou remplacé par x, *.
Les parenthèses '(', ')' ou les crochets '[', ']' peuvent être utilisés dans l'écriture de la fonction.

Expression booléenne : [**(!a + !b + !c)(!b + c)] [(!a + !b + !c)(a!b + b!c + c!a)]**

((!a+!b+!c).(!b+c)).((!a+!b+!c).(a.!b+b.!c +c.!a))	b.c	b.!c	!b.!c	!b.c
a	0	0	1	1
!a	1	0	0	1

Forme normale disjonctive (somme de produits) : !a !b c + !a b c + a !b !c + a !b c

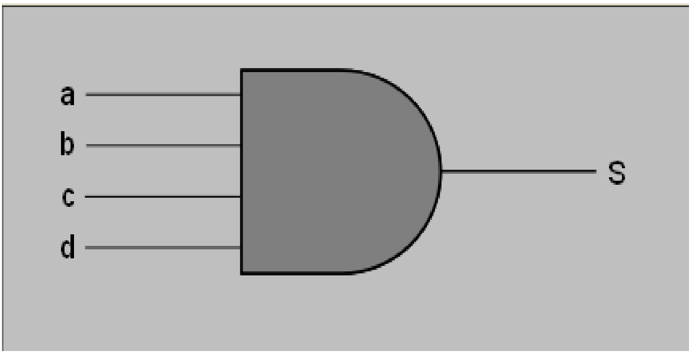
Écriture simplifiée (avec recouvrements éventuels) : a !b + !a c + !b c

IV- PORTES LOGIQUES, CIRCUIT LOGIQUE :

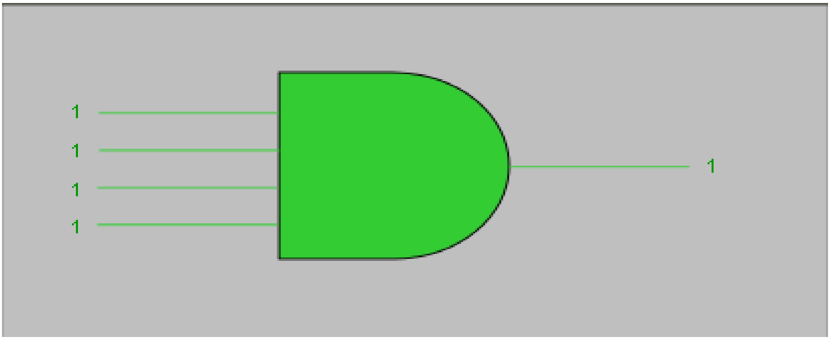
Porte logique «ET» «AND» :

La sortie de la porte « ET » est à 1 si toutes les entrées sont à 1

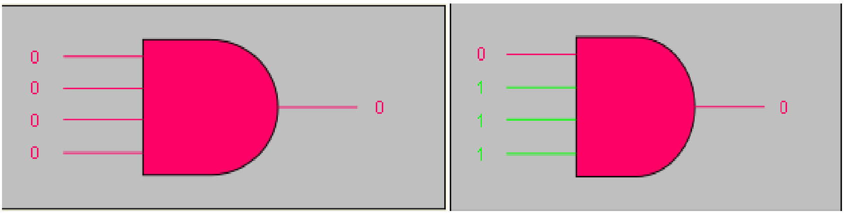
- Sortie ET a 1 si : a ET b ET c ET d sont à 1



- La sortie de la porte «ET» est à 1 si toutes les entrées sont à 1 . Il y a un seul cas



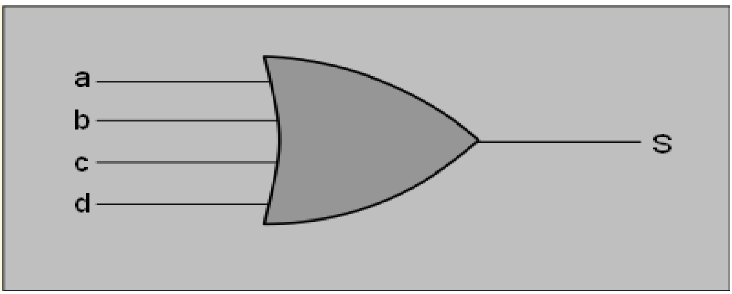
Dans les autres cas la sortie est à '0' :



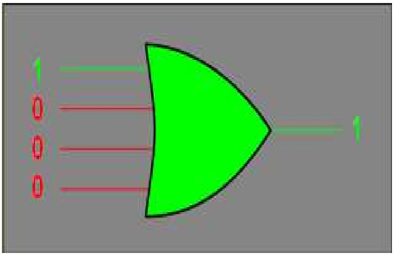
Porte logique «OU»«OR» :

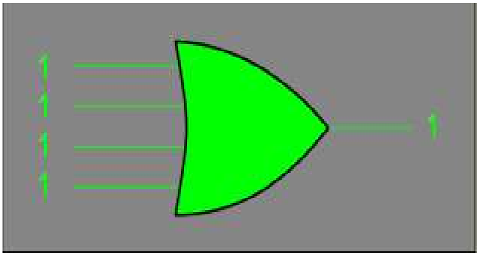
La sortie du circuit «OU» est à 1 si une entrée est à 1.

- sortie OU a 1 si : a OU b OU c OU d est à 1

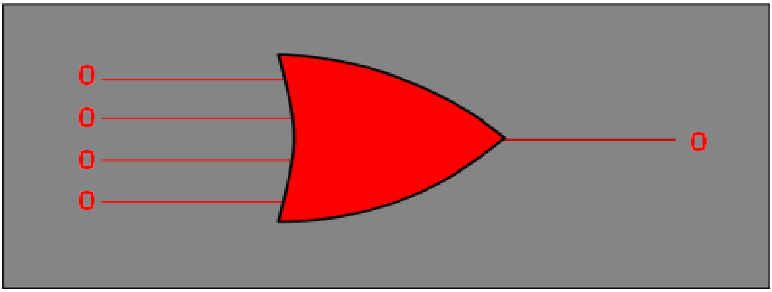


Si une entrée est à 1, la sortie est à 1 :





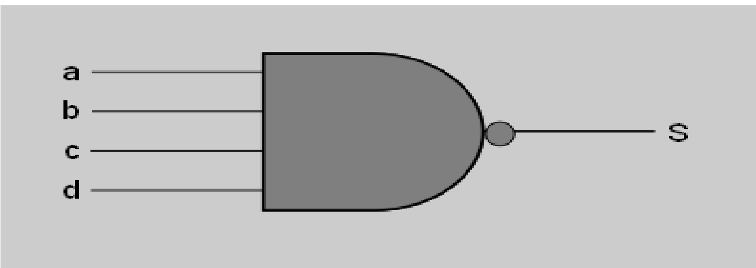
Si toutes les entrées sont à 0 alors la sortie est à 0. Il y a un seul cas :



Porte logique «NON-ET»«NAND» :

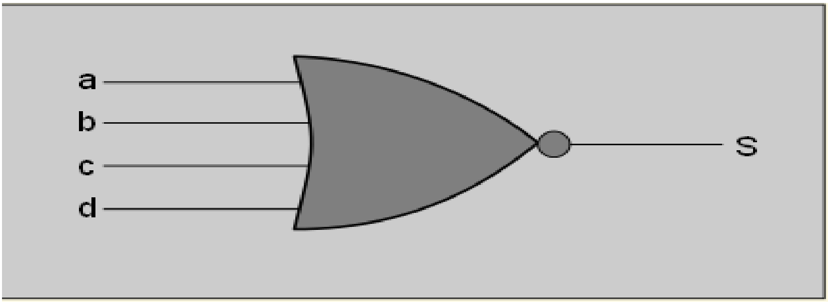
La sortie de la porte « NON-ET » est à 1si une entrée est à 0.

- Sortie NON-ET a 1 si : a OU b OU c OU d est à 0



La sortie du circuit «NON-OU» est à 1 si toutes les entrées sont à 0.

- Sortie NON-OU a 1 si : a ET b ET c ET d sont à 0



1- Circuits logiques :

Circuits ayant un certain nombre d'entrées et de sorties, chacune d'entre elles ne pouvant prendre que 2 états.

2- Circuits logiques combinatoires :

Circuits ayant un certain nombre d'entrées et de sorties, chacune d'entre elles ne pouvant prendre que 2 états

3 -Circuits logiques combinatoires :

L'état de la sortie est une fonction déterminée de l'état des entrées :S = f(a, b,...)

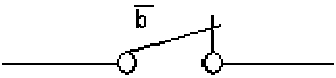
4-Circuits logiques séquentiels :

L'état seul des entrées ne suffit pas pour connaître l'état de la sortie car à un même état des entrées peuvent correspondre des états de sortie différents.

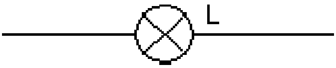
5-Schémas électriques :

Un des premiers types de circuit logique qui vient à l'esprit.
Les symboles utilisés sont

contact normalement ouvert



contact normalement fermé



ampoule

Les contacts sont toujours dessinés au repos.

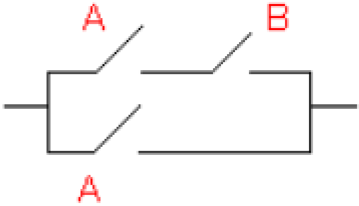
Un contact ouvert au repos est représenté par une lettre minuscule.

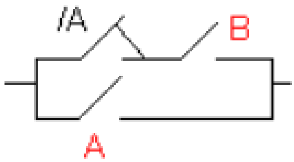
Un contact fermé au repos est représenté par une lettre minuscule surmontée d'une barre.

Les différents organes sont dits à l'état 0 lorsqu'ils sont au repos (non actionnés, non alimentés) et à l'état 1 lorsqu'ils fonctionnent.

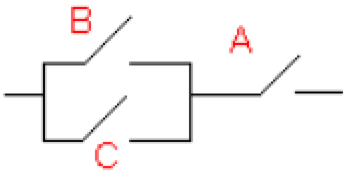
EXEMPLES D'APPLICATIONS :

- A + AB = A

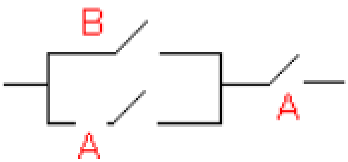




- **A + /AB = A + B**



- **A (B + C) = AB + AC**



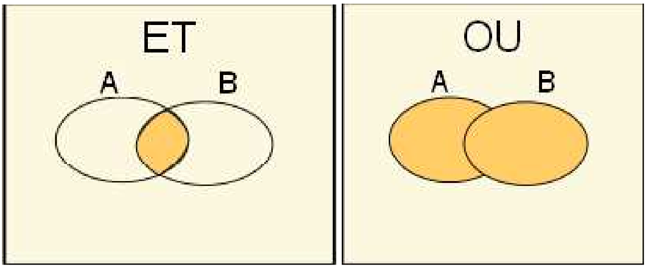
- **A (A + B) = A +AB = A**

V- SIMPLIFICATION DES CIRCUITS LOGIQUES : DIAGRAMME DE KARNAUGH :

1- Diagramme de Venn :

Représentation graphique d'une fonction logique :

ET: intersection OU: réunion



2- Tableau de « KARNAUGH »:

Forme stylisée d'un diagramme de Venn

- Chaque case correspond à :
  - Une combinaison des valeurs des entrées
  - Donc à un minterme
  - Donc à une ligne de la table de vérité

- Dans chaque case on indique la valeur de la fonction :

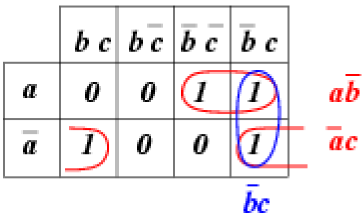
  - 0
  - 1
  - ou Ø => état indifférent

Explications :

L'application construit le tableau de Karnaugh de la fonction booléenne et l'écrit sous forme normale disjonctive de la fonction. En outre l'application donne une expression simplifiée de la fonction.

Le tableau est prévu pour trois variables a, b, c. Si l'une des variables ou plusieurs d'entre elles sont absentes de l'expression, le nombre de cases du tableau et leur position ne changent pas et les trois variables peuvent apparaître dans la forme normale disjonctive.

La forme normale disjonctive est soit 0 soit une expression des trois variables a, b, c. C'est une somme d'au plus huit produits de trois facteurs. Les trois facteurs sont a, b, c ou leurs compléments !a, !b et !c. (Par exemple, un terme pourrait être a!b!c).



Simplification de $a\bar{b} + \bar{a}c + \bar{b}c$

À chacun des termes de la forme normale disjonctive correspond une case du tableau et, inversement, à chaque case du tableau correspond un produit. (Par exemple, a!b!c se trouve dans la case
définie par 'a' à gauche et par b!c! au-dessus).

Lorsque la fonction est constante égale à 0 ou à 1, le programme l'indique, Sinon une écriture simplifiée utilisant les variables a, b ou c est donnée.

Dans certains cas l'expression peut encore se simplifier, voir l'exemple par défaut où a.!b + !a.c + !b.c peut s'écrire a.!b + !a.c. Lorsque le nombre de variables n'est pas élevé, le tableau de Karnaugh permet de simplifier assez simplement les expressions obtenues. L'image ci-contre montre comment simplifier a!b + !ac + !bc en a!b + !ac.

	bc	$b\bar{c}$	$\bar{b}\bar{c}$	$\bar{b}c$
a	1	0	0	1
$\bar{a}$	1	0	0	1

$$c = abc + a\bar{b}c + \bar{a}bc + \bar{a}\bar{b}c$$

Les groupements considérés de plusieurs cases marquées '1' ont

- 8 cases (toutes) : la fonction est égale à la constante '1'
- 4 cases (consécutives* ou en carré*) : le terme correspondant aux 4 cases est formé d'une seule variable ou de son complément
- 2 cases (accolées*) : le terme est composé de deux variables (a !b par exemple).
- 1 case : les termes sont composés de trois variables (ou de leurs compléments)
- 0 case : la fonction est nulle.

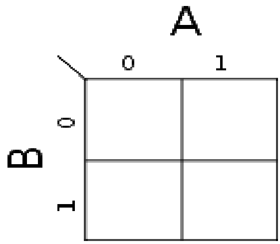
*** : sur le schéma, les deux cases !ac (en rouge) sont considérées accolées, de même dans un autre exemple on considèrerait que les 4 cases c = abc+a!bc+!abc+!a!bc forment un carré**.

Remarques: Lorsque l'expression proposée est mal construite, certaines erreurs sont détectées, une expression vide et un caractère incorrect seront signalés ainsi que certaines erreurs de parenthèses ou de positions des opérateurs.

Si le tableau ne s'affiche pas, c'est que l'expression entrée est incorrecte et que le type de l'erreur n'a pu être déterminé. Si vous voulez obtenir la forme normale conjonctive de f, cherchez la forme disjonctive de !f à l'aide de cette page et déduisez-en f = !(!f) à la main. par exemple avec f = a.(b+c)+!a.!c, calculez !f = !(a.(b+c)+!a.!c) = !a c + a !b !c et déduisez de cette dernière expression f = (a+!c)(!a+b+c) et vérifiez le résultat.

Exemple pour simplifier la construction de Tableau de Karnaugh :

1ère étape : construire le tableau



Si l'on a 2 variables a et b :

- a est placé en colonne,
- b sera placé en ligne,

Ce qui donne le tableau suivant :

De ce fait, on se retrouve avec 4 cases vides, correspondant aux 4 combinaisons possibles de 2 variables pouvant prendre 2 états.

Avec notre exemple, Voilà ce que donne le tableau rempli :

F		a	
		0	**1**
b	**0**	0	1
	1	1	1

À partir du moment où le tableau est correctement rempli, on peut commencer à résoudre le système.

2ème étape *:* Rassemblement

Le but est très simple. Il faut effectuer des regroupements de 1 ou de X ; par paquets de 1, 2, 4 ,8, 16....2^n. Ces regroupements doivent être des rectangles ou des carrés, jamais de travers, et les plus grands possible sachant qu'un élément déjà utilisé peut être repris. **Attention :** ne pas oublier que le tableau de Karnaugh est *écrit* sur un cylindre.

Avec notre exemple : on peut faire ces 2 rassemblements :

F		a	
		0	1
b	0	0	1
	1	1	1

F		a	
		0	1
b	0	0	1
	1	1	1

3ème étape : Résolution des rassemblements

Maintenant que l'on a pu, avec différentes sélections, prendre tous les 1, on essaie de résoudre les rassemblements. Pour cela, il faut que les variables participants au rassemblement concerné ne changent pas.

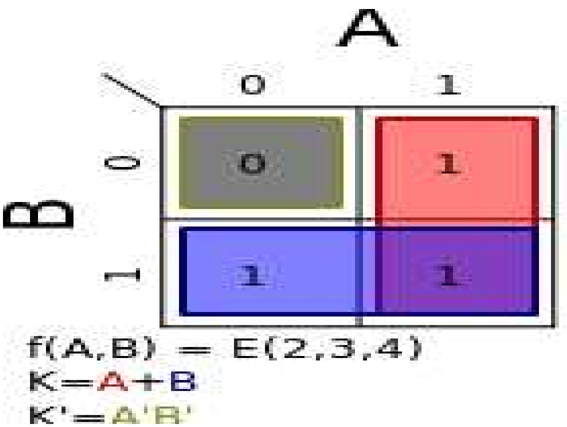
- Avec le rassemblement vert :

  La variable a est toujours à 1
  La solution verte donne : a.

- Avec le rassemblement rouge :

  La variable b est toujours à 1
  La solution rouge donne : b.

La solution finale du tableau est : $F = a + b$



Exemples :

- **Exemples à 0 ou 1 variable**

  0 + 0, 0 + 1, 1 + 1, 0 . 0, 0 . 1, 1 . 1,

  a + 0, a + 1, a . 0, a . 1, a + a, a . a, a + !a, a . !a,

  Exemples à 2 variables

  Commutativité b + a, b . a,
  Compléments !(a + b), !(a . b), a.b + a.!b,
  Absorption a + a.b,

- **Exemples à 3 variables**

  Distributivités a (b + c), (a + b) . (a + c),

  Produits de sommes (a + !b) . (b + !c),
  (a+!b).(b+!c)(c+!a),

  (a+b+c)(!a+b)(!b+c)(!c+a), (a+b+c) . (!a+!b+!c),

  (!a+b+c) . (a+!b+c). (a+b+!c),

Exemples :

Donner la Négations de ces sommes dans le tableau de Karnaugh

- !(a!b + b!c + c!a), !(a!bc + ab!c + !abc), !(ab + bc + ac+a!b),

Tableau de Karnaugh

!(a.!b+b.!c+c.!a)	**b.c**	**b.!c**	**!b.!c**	**!b.c**
a	**1**	**0**	**0**	**0**
!a	**0**	**0**	**1**	**0**

Forme normale disjonctive (somme de produits) : !a !b !c + a b c

Écriture simplifiée (avec recouvrements éventuels) : a b c + !a !b !c

- !(a!bc + ab!c + !abc)

Tableau de Karnaugh

!(a.!b.c+a.b.!c+!a.b.c)	b.c	b.!c	!b.!c	!b.c
a	**1**	**0**	**1**	**0**
!a	**0**	**1**	**1**	**1**

Forme normale disjonctive (somme de produits) : !a !b !c + !a !b c + !a b !c + a !b !c + a b c

Écriture simplifiée (avec recouvrements éventuels) : !a !c + !a !b + !b !c + a b c

- !(ab + bc + ac+a!b)

Tableau de Karnaugh

!(a.b+b.c+a.c+a.!b)	b.c	b.!c	!b.!c	!b.c
a	**0**	**0**	**0**	**0**
!a	**0**	**1**	**1**	**1**

Forme normale disjonctive (somme de produits) : **!a !b !c + !a !b c + !a b !c**

Écriture simplifiée (avec recouvrements éventuels) : **!a !c + !a !b**

RÉFÉRENCE WEBOGRAPHIQUE :

http://www.gilu.net/introduction_a_l_algebre_de_boole.htm

http://www.fsr.ac.ma/cours/informatique/mouline/AlgBoole.pdf

http://mathinfovannes.free.fr/articles.php?lng=fr&pg=130

http://www.reds.ch

http://fr.wikiversity.org/wiki/Logique_de_base